Principled Technologies®

TEST REPORT
MAY 2010

Effects of congestion on TCP transactions: comparison of Cisco Nexus 5010 and Arista 7124S network switches under incast conditions

Executive summary

Cisco Systems®, Inc. (Cisco) commissioned Principled Technologies (PT) to examine how well two network switches, the Arista 7124S and the Cisco Nexus 5010, handle a TCP traffic workload that can lead to a severe impact on TCP transmission rates and transaction times, conditions referred to as *incast*.

Modern network switches provide buffer space that holds incoming network packets on ingress or egress queues during congestion. As the volume of incoming data increases in packet size and rate, the switch buffers fill and experience packet drop until they can store no more data. This causes the TCP throughput collapse known as incast (http://www.pdl.cmu.edu/Incast/).

KEY FINDINGS

- Switch buffering approaches can severely affect TCP transmission rates when an incast condition occurs.
- During an incast condition, the Arista7124S had a 134.9% slower transaction completion time than the Cisco Nexus 5010. (See Figure 1.)
- The cause of this completion time difference was 3.4 times as many dropped TCP packets on the Arista 7124S as on the Cisco Nexus 5010 (see Figure 4), which in turn led to retransmissions.

As the CMU paper we cited above describes, the problem arises because a client requests data blocks from multiple sources simultaneously, which together send enough data to cause congestion, forcing the ingress buffer to queue packets. After exceeding the ingress buffer depth, the switch drops packets, which leads to TCP timeouts and retransmissions. A TCP packet drop triggers TCP slow-start, further contributing to reduced goodput (which we define below) and timeouts, which force TCP retransmissions, thus adding to traffic volume unnecessarily. The lower goodput causes longer transaction times.

The traffic patterns that lead to incast conditions happen in environments such as cluster-based and iSCSI storage systems, as well as some Web 2.0 applications. What these environments all share is a system requesting data simultaneously from multiple other systems. For more information on this problem and its technical underpinnings, see http://www.eecs.berkeley.edu/~ychen2/professional/TCPIncastWREN2009.pdf.

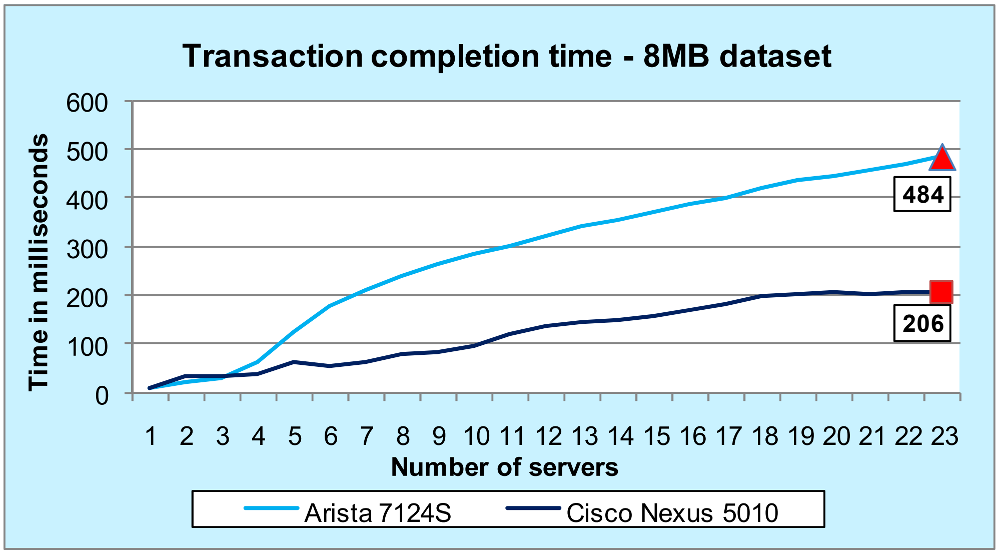

Figure 1: Transaction completion time, in milliseconds, using an 8MB dataset. Lower numbers are better.

To learn more about incast and its effect on application performance with different network platforms, we compared the following two 24-port, 10G Ethernet switches: the Arista 7124S and the Cisco Nexus 5010.

Figure 1 shows the average transaction completion time, in milliseconds, for the Arista 7124S and the Cisco Nexus 5010 using an 8MB dataset and a maximum transmission unit (MTU) setting of 9,000 bytes. We performed 1,000 iterations for testing. The chart shows the average transaction completion time for those 1,000 iterations. Lower completion times are better.

NOTE

Another concept directly related to transaction completion time is goodput. Goodput is the amount of useful bits, in Mbps, transmitted through the network switch to the application. As a network switch packet drops increase, transactions take longer to complete, and so the goodput of the transaction decreases. Consequently, lower transaction completion times increase the application goodput.

Workload

The test environment uses a workload in which a single client requests data from multiple servers replicating the behavior of cluster-based storage systems. Berk Atikoglu and Tom Yue, of the Department of Electrical Engineering at Stanford University, created a program that runs such a workload. (http://www.stanford.edu/~atikoglu/r2d2/) It is based on the Carnegie Mellon University incast research (http://www.pdl.cmu.edu/Incast/), which can recreate an incast condition. We used the program to measure the transaction completion time of the two switches. We were also able to measure dropped packets, retransmissions, and goodput. The test has one receiver (client) making a request for a certain block size of data from a set of senders (servers). Each server sends the requested data to the client. Figure 2 illustrates how the switch distributes data across the servers during an example test of four servers.

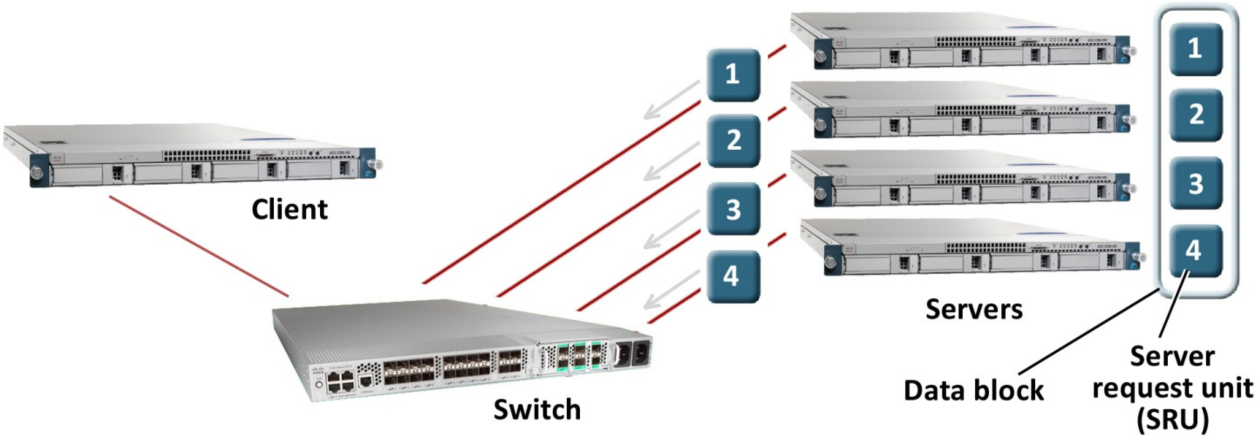

Figure 2: Illustration of the incast workload data transfer.

The test iteratively increases the number of servers from one to n, where n is the total number of servers in the test bed (23 for this test). The results provide goodput, TCP timeouts, TCP retransmissions and packet drop.

Test results

For testing, we used three different sized datasets to simulate different kinds of data transmitted over a typical production network. We also used different MTU sizes. We tested the following three configurations:

- 8MB dataset, 9,000 MTU (non-striped)
- 4MB dataset, 9,000 MTU (non-striped)
- 10MB dataset, 1,500 MTU (striped)

The non-striped configurations request either 8MB or 4MB of data from each server. For example, in a two-server configuration testing an 8MB dataset, each server transmits 8 MB of data for a total of 16 MB of data received. In the striped configuration, the application on the client requests a total of 10 MB of data striped across the two servers, each transmitting 5 MB of data, for a total of 10 MB of data received.
For each configuration, we began the test using a single server. We then added one server and repeated the test. We continued adding servers one at a time, to a maximum of 23 servers. The results show how each network switch handled the increasing load. See Appendix C for complete results.

Here, we briefly describe the metrics we report:

- **Transaction completion time -** The total time, in milliseconds, the test takes to complete one test iteration. We performed 1,000 iterations and averaged the transaction completion time across all of them. Lower numbers, representing faster completion times, are better.
- **Goodput -** The number of useful bits, in Mbps, transmitted over the network as seen by the application on the system. Higher numbers are better.
- **Dropped packets -** The number of packets dropped due to congestion and network switch buffer overflow conditions. Lower numbers are better.
- **TCP retransmissions -** The number of times the sending servers must resend packets. Lower numbers are better.

When reviewing both transaction completion time and goodput, one must take in to account the number of dropped packets and retransmissions. As the levels of congestion increase and buffer overflow conditions exacerbate, packet drop increases and TCP timeouts and retransmissions increase on a similar scale.

Figure 3 shows the dropped packets for the 8MB dataset, 9,000 MTU configuration. As the number of servers increase, the incast condition becomes more severe. We ran 1,000 iterations of each dataset for testing. The dropped packets and TCP retransmission results display a single iteration, which is the average of the 1,000 iterations.

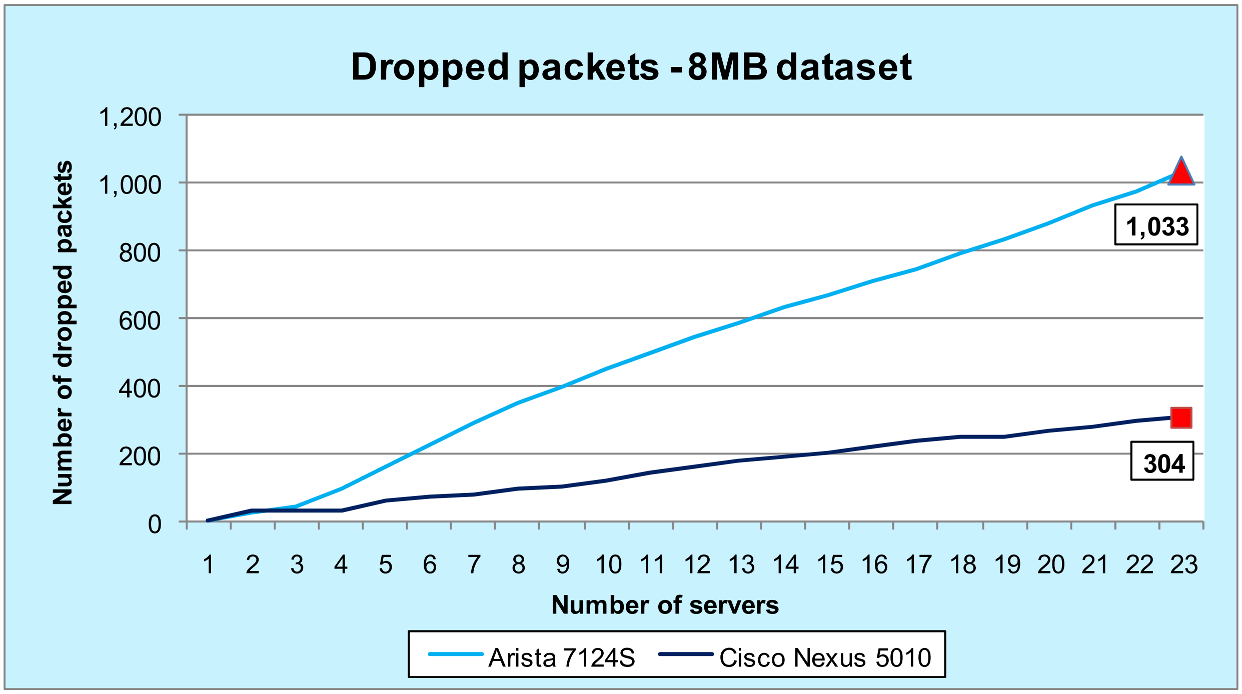

Figure 3: Number of dropped packets for the Arista 7124S and Cisco Nexus 5010 using an 8MB dataset. Lower numbers are better.

As we expected, the dropped packets lead to a similar number of TCP retransmissions, as Figure 4 illustrates.

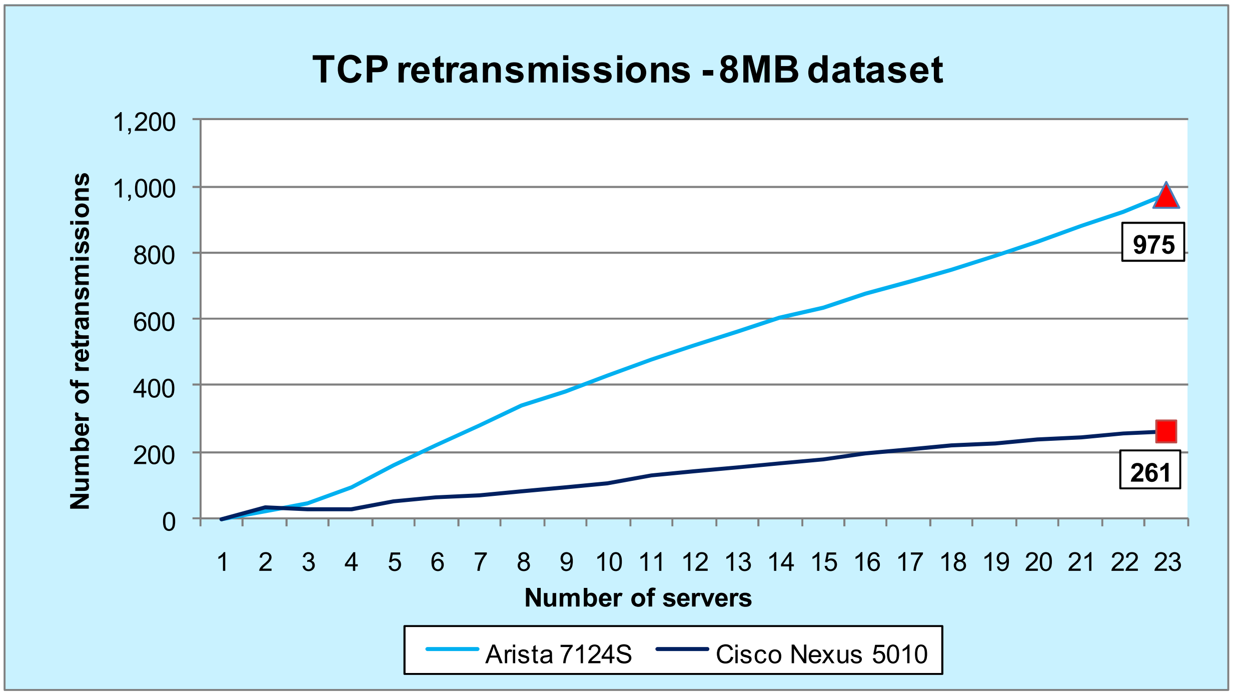

Figure 4: Number of TCP retransmissions for the Arista 7124S and Cisco Nexus 5010 using an 8MB dataset. Lower numbers are better.

The TCP timeouts trigger retransmissions as well as the TCP congestion avoidance slow-start algorithm. The reduced outstanding window size can compound the effect on goodput lowering the effective data transfer rate. Figure 5 shows the goodput for the 8MB dataset, 9,000 MTU configuration. The goodput is the amount of data, in Mbps, each switch provides while dealing with incast. Higher numbers are better.

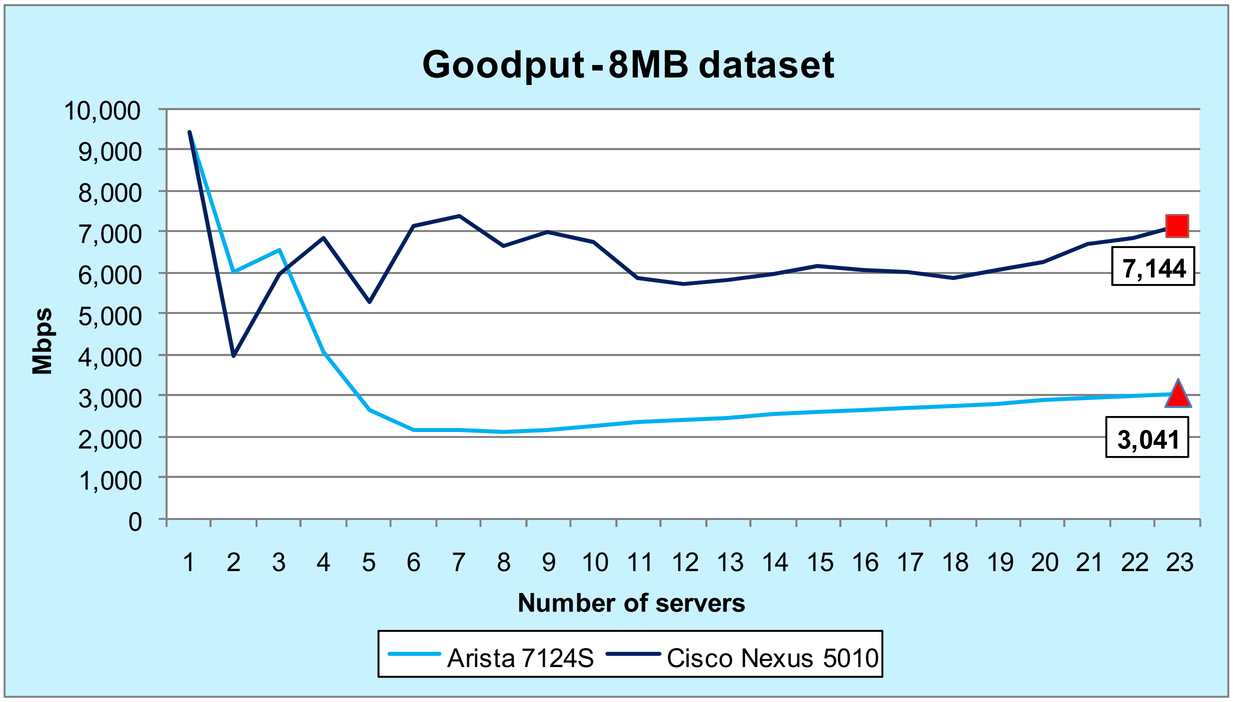

Figure 5: Goodput (in Mbps) for the Arista 7124S and Cisco Nexus 5010 using an 8MB dataset. Higher numbers are better.

We also looked at a smaller dataset size, while keeping the same MTU size, to see what happened and how incast levels and switch performance varied. Figure 6 shows the transaction completion time for both switches running a 4MB, 9,000MTU dataset. While the dataset is smaller, we saw similar behavior as the 8MB dataset. The Arista7124S had a transaction completion time of 379 milliseconds with 23 servers, which is 168.8% slower transaction completion time than the Cisco Nexus 5010. The Cisco Nexus 5010 had a transaction completion time of 141 milliseconds with 23 servers. The incast condition occurred, causing the buffers to overflow and packet drops. As with the 8MB dataset, TCP timeouts and TCP retransmissions took place, which in turn made the completion time increase as the traffic load increased.

As with the 8MB dataset, we show the transaction completion time for a single 4MB dataset in milliseconds. The single transaction is the average of 1,000 iterations.

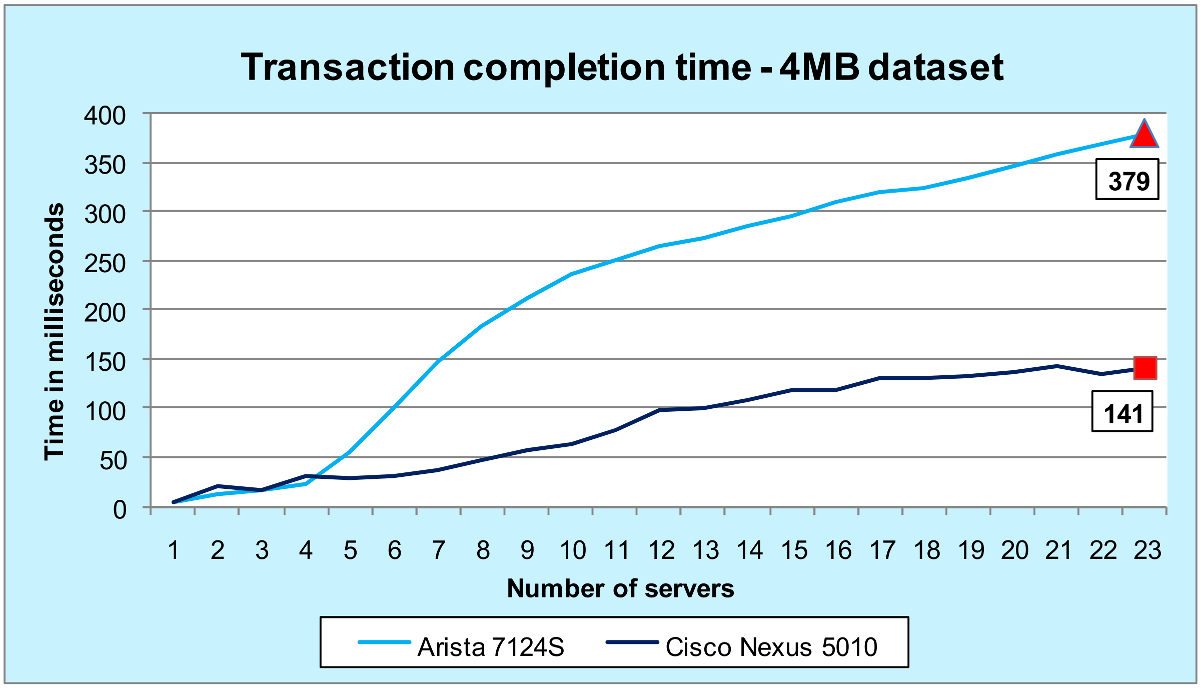

Figure 6: Transaction completion time, in milliseconds, for the Arista 7124S and Cisco Nexus 5010 using a 4MB dataset. Lower numbers are better.

We also tried using a smaller MTU that might be more typical of some Web 2.0 applications, as well as striping the data across the servers. For this configuration, we used a 10MB dataset. Figure 7 shows the transaction completion time for the 10MB dataset, 1,500 MTU configuration with data striping. Note that, because striping divides the total data by the number of servers, the amount of data received by the client is constant for all the data points. This yields the differently shaped curve in Figure 7.

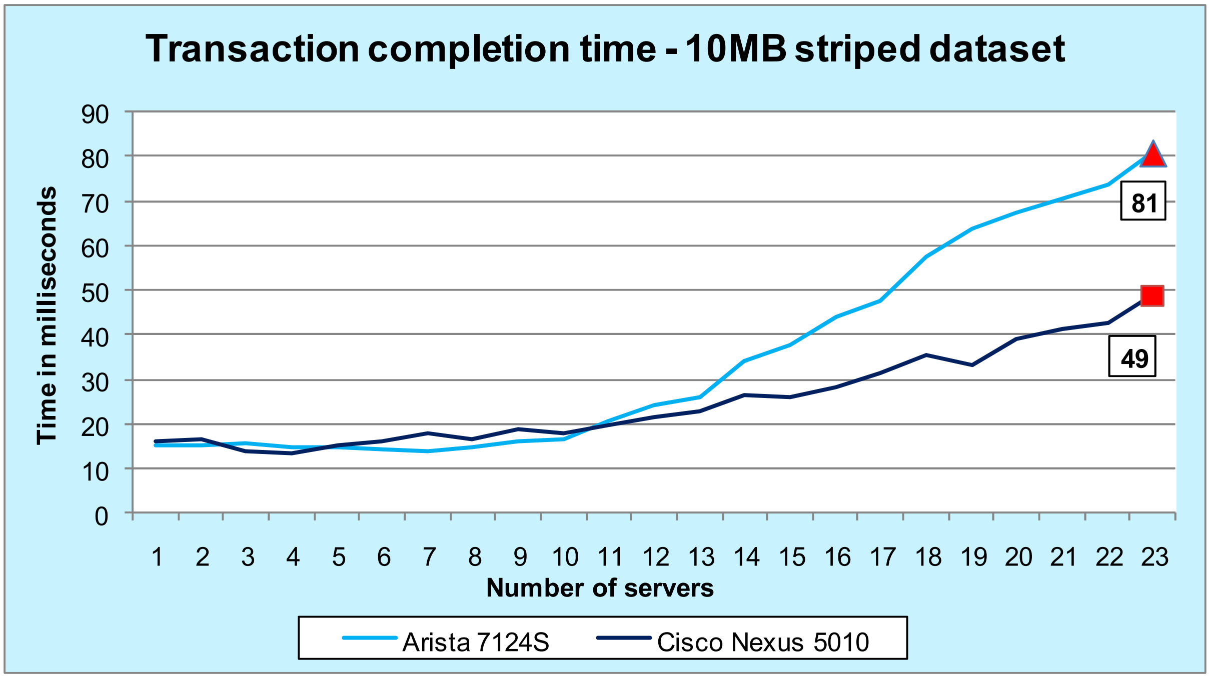

Figure 7: Transaction completion time, in milliseconds, for the Arista 7124S and Cisco Nexus 5010 using a 10MB striped dataset. Lower numbers are better.

Conclusion

As more environments use cluster-based and iSCSI storage systems as well as Web 2.0 applications (where a system requests data simultaneously from multiple other systems), the possibility of incast occurring will increase. PT's testing showed the impact of incast on the Arista 7124S and the Cisco Nexus 5010 24-port, 10G Ethernet switches. The buffering of the Cisco Nexus 5010 allowed it to better handle the incast condition. The Cisco Nexus 5010's buffering dropped fewer packets, which led to fewer TCP retransmissions and thus higher goodput and lower transaction completion times.

Test methodology

We tested the switches at Cisco's facility. Cisco supplied the two switches as well as the 24 servers we used in the test bed. Figure 8 illustrates the test bed.

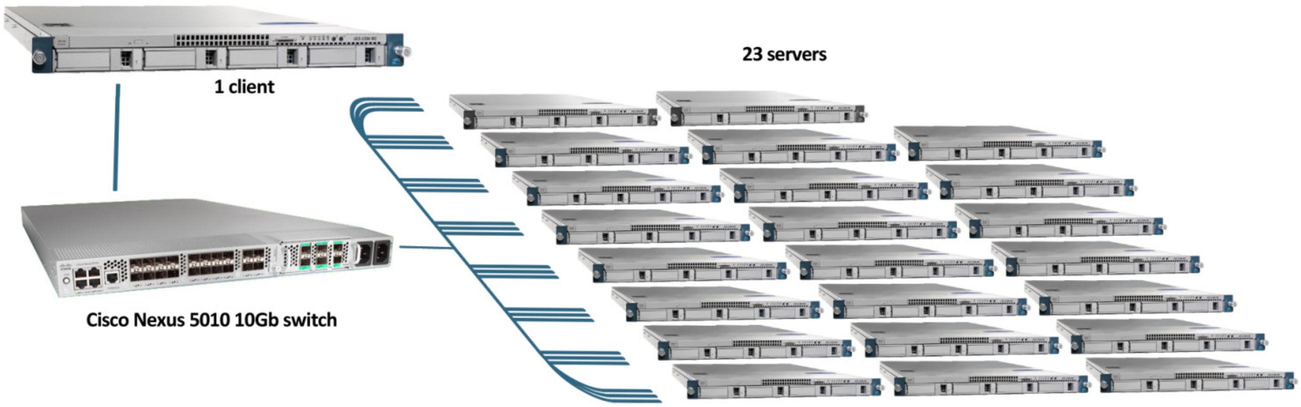

Figure 8: The test bed we used.

Test bed setup

We used 24 Cisco UCS C200 M1 servers for testing. Each server had an Intel® NetEffect™ NE020 Server Cluster Adapter installed. We connected each server to the switch via a 10GB SFP cable. We used one of the systems as the test client and the other 23 as test servers. We configured all servers with Red Hat® Enterprise Linux® 5.4, kernel-2.6.18-164.el5. We used the default installation options, but disabled SELinux and Firewall.

Prior to testing, we made sure the Arista 7124S and Cisco Nexus 5010 were using the latest software and firmware. We configured both switches to use jumbo frames during testing. To adjust the MTU size, we ran the following command on all systems depending on which MTU size we were testing:

- `ifconfig eth2 mtu 1500`
- `ifconfig eth2 mtu 9000`

The test workload had executables for the servers and client. Prior to beginning the test, we started the executable on all servers, which then stayed idle waiting for the client to begin the test. To begin the test, we started the executable on the client, which started the test. The client executable referenced a .dat file, which gives the client information on how the test will run. The .dat file gives information on the number of servers to be in the test, their IP addresses, the requested data size, and the number of iterations to run.

Upon completion, the incast test displays the following results: total data sent, total duration, and goodput in Mbps. To simplify testing and data collection, we used scripts to run all tests and copy results into output files.

Appendix A – Network switch configuration information

Figure 9 provides detailed configuration information about the network switches.

Network switch	Arista 7124S	Cisco Nexus 5010
Hardware version	06.02	v1.2
Software version	4.4.0	4.2 (1) N1 (1) build 0.293
Internal build version	4.4.0-241057.EOS440bugFix	1.2.0 (BIOS version)

Figure 9: Detailed configuration information for the network switches.

Appendix B – Server configuration information

Figure 10 provides detailed configuration information about the test servers.

Servers	Cisco UCS C200 M1
General processor setup	
Number of processor packages	2
Number of cores per processor package	4
Number of hardware threads per core	2
CPU	
Vendor	Intel
Name	Xeon X5540
Stepping	D0
Socket type	LGA1366
Core frequency (GHz)	2.53
Bus speed (GT/s)	5.86
L1 cache (KB)	32 + 32 (per core)
L2 cache (KB)	256 (per core)
L3 cache (MB)	8
Thermal design power (TDP, in watts)	80
Platform	
Vendor and model number	Cisco UCS C200 M1
Motherboard chipset	Intel 5500
BIOS name and version	C200M1.0036.1.0.3.112020091824
BIOS settings	Default
Memory modules	
Vendor and model number	Micron MT36JSZF51272PZ-1G4F1
Size (GB)	4
Number of RAM modules	4
Total RAM in system (GB)	16
Type	PC3-8500
Speed (MHz)	1,066
Speed in the system currently running @ (MHz)	1,066
Timing/Latency (tCL-tRCD-iRP-tRASmin)	7-7-7-13
Chip organization	Double-sided
Hard disk	
Vendor and model number	Seagate ST3146356SS
Number of disks in system	2
Size (GB)	146
Buffer size (MB)	16
RPM	15,000
Type	SAS 3Gb/s
Controller	LSI MegaRAID SAS 8708EM2 PCIe RAID Controller
Operating system	
Name	Red Hat Enterprise Linux 5.4

Servers	Cisco UCS C200 M1
Kernel	2.6.18-164.el5
Language	English
Network card/subsystem	
Vendor and model number	Dual Port Gigabit NIC
Type	PCI Express
Additional network card/subsystem	
Additional NIC	NetEffect NE020 SFP+ SR
Type	PCI Express

Figure 10: Detailed configuration information for the test servers.

Appendix C – Detailed results

Figures 11 through 13 provide detailed results for both switches for each configuration.

4MB dataset, 9,000 MTU configuration

Number of servers	Goodput (Mbps)		Dropped packets		TCP retransmissions		Transaction completion time (milliseconds)	
	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010
1	9,044.0	9,033.8	0.0	0.0	0.0	0.0	3.5	3.5
2	5,456.0	3,213.8	11.3	16.7	11.3	16.6	11.7	19.9
3	6,138.0	5,531.5	18.5	17.6	18.2	16.9	15.6	17.4
4	5,565.6	4,130.0	32.7	20.1	32.0	19.2	23.0	31.0
5	2,946.5	5,760.5	75.0	29.6	73.0	26.8	54.3	27.8
6	1,913.9	6,298.4	125.8	37.7	122.0	33.8	100.3	30.5
7	1,520.6	6,235.2	170.5	40.6	164.6	36.3	147.3	35.9
8	1,396.7	5,493.3	211.9	48.8	204.4	42.5	183.3	46.6
9	1,362.1	4,995.2	258.7	53.6	248.7	47.7	211.4	57.7
10	1,357.3	5,086.3	288.0	59.6	276.4	53.6	235.8	62.9
11	1,401.4	4,519.1	324.9	66.7	310.8	58.7	251.2	77.9
12	1,451.9	3,890.9	352.2	76.5	335.9	66.7	264.5	98.7
13	1,525.5	4,134.5	382.1	81.6	363.8	71.8	272.7	100.6
14	1,570.6	4,168.0	402.3	88.8	382.3	77.6	285.3	107.5
15	1,621.0	4,049.7	433.9	97.7	410.8	84.7	296.1	118.5
16	1,658.2	4,326.3	460.5	105.3	435.3	91.0	308.8	118.4
17	1,704.4	4,161.7	482.4	116.1	455.1	101.3	319.2	130.7
18	1,778.5	4,427.9	507.7	116.9	478.8	104.1	323.9	130.1
19	1,817.3	4,581.7	530.5	123.3	499.6	110.1	334.6	132.7
20	1,848.1	4,658.3	556.9	137.3	523.4	120.5	346.3	137.4
21	1,879.6	4,727.0	580.0	150.4	544.8	129.7	357.5	142.2
22	1,914.7	5,248.0	605.8	152.4	567.9	130.8	367.7	134.2
23	1,941.2	5,237.0	628.5	155.1	589.2	132.6	379.1	140.5

Figure 11: Complete results for 23 servers with the 4MB dataset. The results are the average of 1,000 iterations.

8MB dataset, 9,000 MTU configuration

Number of servers	Goodput (Mbps)		Dropped packets		TCP retransmissions		Transaction completion time (milliseconds)	
	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010
1	9,421.4	9,413.0	0.0	0.0	0.0	0.0	6.8	6.8
2	6,000.5	3,948.8	21.2	31.9	21.0	31.7	21.3	32.4
3	6,539.8	5,970.8	44.5	30.2	43.7	29.2	29.4	32.2
4	4,054.4	6,845.1	96.7	31.2	94.5	29.8	63.1	37.4
5	2,622.4	5,279.0	161.4	58.1	157.4	52.5	122.0	60.6
6	2,172.4	7,134.9	226.9	68.6	220.5	61.9	176.8	53.8
7	2,137.1	7,397.7	290.4	76.6	281.6	68.5	209.6	60.6
8	2,128.4	6,661.3	349.0	95.7	337.6	83.9	240.6	76.9
9	2,180.4	6,992.8	397.7	102.5	383.9	92.3	264.2	82.4
10	2,247.0	6,753.1	447.6	116.4	430.8	104.9	284.8	94.8
11	2,345.5	5,845.8	494.1	144.3	475.0	127.1	300.2	120.4
12	2,405.1	5,721.8	540.8	159.2	519.0	140.0	319.3	134.2
13	2,441.6	5,826.0	585.6	174.9	560.6	153.7	340.8	142.8
14	2,529.1	5,965.6	629.1	188.4	601.6	164.9	354.3	150.2
15	2,579.7	6,148.2	666.5	203.4	636.1	177.1	372.1	156.1
16	2,644.3	6,075.6	708.1	221.1	674.7	192.6	387.3	168.5
17	2,715.4	5,989.9	745.1	236.8	709.1	208.2	400.7	181.6
18	2,750.1	5,851.7	787.8	246.2	748.0	221.2	418.9	196.9
19	2,795.4	6,058.2	834.2	250.7	791.1	225.2	435.0	200.7
20	2,880.9	6,266.1	876.2	266.2	830.0	234.7	444.3	204.3
21	2,944.9	6,697.9	930.0	278.9	880.0	242.4	456.4	200.7
22	2,988.9	6,822.8	973.4	295.3	920.0	254.7	471.1	206.4
23	3,041.3	7,144.7	1,032.8	304.1	975.0	261.2	484.0	206.0

Figure 12: Complete results for 23 servers with the 8MB dataset. The results are the average of 1,000 iterations.

10MB dataset, 1,500 MTU configuration (striped)

Number of servers	Goodput (Mbps)		Dropped packets		TCP retransmissions		Transaction completion time (milliseconds)	
	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010	Arista 7124S	Cisco Nexus 5010
1	5,333.4	5,018.5	0.0	0.0	4.3	15.0	15.0	15.9
2	5,200.7	4,845.1	3.9	10.5	4.3	10.4	15.4	16.5
3	5,176.4	5,737.6	17.3	40.7	18.0	40.1	15.5	13.9
4	5,470.9	5,987.8	27.2	31.9	27.3	30.9	14.6	13.4
5	5,378.0	5,234.1	41.2	24.6	42.1	31.3	14.9	15.3
6	5,580.1	4,918.3	49.3	17.7	50.8	31.9	14.3	16.3
7	5,859.8	4,513.2	62.4	10.8	62.7	34.1	13.7	17.7
8	5,466.9	4,842.7	78.1	8.9	80.1	41.1	14.6	16.5
9	4,919.0	4,240.3	84.9	7.4	88.7	52.3	16.3	18.9
10	4,902.0	4,484.3	98.0	4.8	101.5	57.0	16.3	17.8
11	3,866.8	4,025.7	112.8	4.3	115.8	64.1	20.7	19.9
12	3,279.4	3,725.6	122.5	4.0	127.4	70.5	24.4	21.5
13	3,064.2	3,498.7	139.7	2.6	142.9	74.4	26.1	22.9
14	2,341.6	3,039.1	148.6	2.1	154.2	79.8	34.2	26.3
15	2,124.5	3,085.4	156.4	1.5	162.6	82.3	37.7	25.9
16	1,818.1	2,847.1	164.5	1.5	172.3	88.9	44.0	28.1
17	1,681.0	2,537.1	175.3	0.9	183.3	92.8	47.6	31.5
18	1,387.9	2,270.7	181.7	1.2	192.1	97.7	57.6	35.2
19	1,250.9	2,399.1	186.4	0.5	199.3	98.4	64.0	33.4
20	1,187.3	2,044.3	184.2	0.4	198.6	101.5	67.4	39.1
21	1,134.6	1,937.5	196.2	0.6	209.7	104.1	70.5	41.3
22	1,083.8	1,883.5	186.9	0.4	205.8	109.3	73.8	42.5
23	990.2	1,641.0	184.8	0.4	204.9	113.7	80.8	48.8

Figure 13: Complete results for 23 servers with the 10MB striped dataset. The results are the average of 1,000 iterations.

About Principled Technologies

We provide industry-leading technology assessment and fact-based marketing services. We bring to every assignment extensive experience with and expertise in all aspects of technology testing and analysis, from researching new technologies, to developing new methodologies, to testing with existing and new tools.

When the assessment is complete, we know how to present the results to a broad range of target audiences. We provide our clients with the materials they need, from market-focused data to use in their own collateral to custom sales aids, such as test reports, performance assessments, and white papers. Every document reflects the results of our trusted independent analysis.

We provide customized services that focus on our clients' individual requirements. Whether the technology involves hardware, software, Web sites, or services, we offer the experience, expertise, and tools to help you assess how it will fare against its competition, its performance, whether it's ready to go to market, and its quality and reliability.

Our founders, Mark L. Van Name and Bill Catchings, have worked together in technology assessment for over 20 years. As journalists, they published over a thousand articles on a wide array of technology subjects. They created and led the Ziff-Davis Benchmark Operation, which developed such industry-standard benchmarks as Ziff Davis Media's Winstone and WebBench. They founded and led eTesting Labs, and after the acquisition of that company by Lionbridge Technologies were the head and CTO of VeriTest.

Principled Technologies, Inc.
1007 Slater Rd., Suite 300
Durham, NC 27703
www.principledtechnologies.com
info@principledtechnologies.com

Disclaimer of Warranties; Limitation of Liability:
PRINCIPLED TECHNOLOGIES, INC. HAS MADE REASONABLE EFFORTS TO ENSURE THE ACCURACY AND VALIDITY OF ITS TESTING, HOWEVER, PRINCIPLED TECHNOLOGIES, INC. SPECIFICALLY DISCLAIMS ANY WARRANTY, EXPRESSED OR IMPLIED, RELATING TO THE TEST RESULTS AND ANALYSIS, THEIR ACCURACY, COMPLETENESS OR QUALITY, INCLUDING ANY IMPLIED WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE. ALL PERSONS OR ENTITIES RELYING ON THE RESULTS OF ANY TESTING DO SO AT THEIR OWN RISK, AND AGREE THAT PRINCIPLED TECHNOLOGIES, INC., ITS EMPLOYEES AND ITS SUBCONTRACTORS SHALL HAVE NO LIABILITY WHATSOEVER FROM ANY CLAIM OF LOSS OR DAMAGE ON ACCOUNT OF ANY ALLEGED ERROR OR DEFECT IN ANY TESTING PROCEDURE OR RESULT.

IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC. BE LIABLE FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH ITS TESTING, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC.'S LIABILITY, INCLUDING FOR DIRECT DAMAGES, EXCEED THE AMOUNTS PAID IN CONNECTION WITH PRINCIPLED TECHNOLOGIES, INC.'S TESTING. CUSTOMER'S SOLE AND EXCLUSIVE REMEDIES ARE AS SET FORTH HEREIN.